

# The Physics of Evolving Complex Software Systems

Prof. T V Gopal

Department of Computer Science and Engineering, Anna University, Chennai – 600 025, India

e-mail: [gopal@annauniv.edu](mailto:gopal@annauniv.edu)

---

**Abstract-** Software has been the adhesive for many multidisciplinary systems. The core engineering disciplines have been banking on the well formulated design principles unlike Software. Yet, Software Engineers celebrate complexity. This paper is on exploring the physics of Cyber - Physical Systems. It is based on mathematical foundations and positions the analog systems in the digital space of the evolving dynamics of design, development and use.

---

**Keywords:** *Cyber – Physical Systems, Complexity, Infinity, Infinitesimals, Computing, Computational Models*

---

## I. INTRODUCTION

"According to Turing-Church, all physically realizable dynamics are equivalent to computation..."

- Michael Conrad

"Software is like entropy. It is difficult to grasp, weighs nothing, and obeys the second law of thermodynamics; i.e. it always increases."

- Norman R. Augustine

"It would be fatuous to assume that every man is constantly aware of the details of his surroundings. I do not believe this to be true. But I am convinced that a well-set dinner table will aid the flow of gastric juices; that a well-lighted and planned classroom is conducive to study; that carefully selected colors chosen with an eye to psychological influence will develop better and more lucrative work habits for the man at the machine; that a quietly designed conference room at the United Nations headquarters might well help influence the representatives to make a calm and just decision. I believe that man achieves his tallest measure of serenity when surrounded by beauty. We find our most serene moments in great cathedrals, in the presence of fine pictures and sculpture, on a university campus, or listening to magnificent music. Industry, technology, and mass production have made it possible for the average man to surround himself with this serenity in his home and in his place of work. Perhaps it is this serenity which we need most in the world, today."

- Henry Dreyfuss, *Designing for People*, 1955

Evolving Complex Software Systems have thus made their design an intriguing blend of Art, Science, Discipline and the Psychology of the stakeholders who make them by design. The specifications of these systems are invariably replete with contradictions, ambiguities, vagueness, incompleteness and mixed levels of abstraction. The true challenge is to design and develop collaborative systems around notions of space which mimic the spatial organization of the real world and then support the emergent patterns of human behavior and interaction which are exhibited by everyday actions in the physical world.

“For the mathematical technician these complications may provide a useful discipline. The person who wants to understand the place of mathematics in modern civilization is merely distracted and disheartened by them. What follows is for those who have been already disheartened and distracted, and have consequently forgotten what they may have learned already, or fail to see the meaning or usefulness of what they remember. So we shall begin at the very beginning”.

- **Lancelot Hogben, Mathematics for the Million, 1936**

Konrad Zuse was the first to propose that physics is just computation, suggesting that the history of our universe is being computed on, say, a cellular automaton. His "**Rechnender Raum**" (Computing Cosmos / Calculating Space) started the field of Digital Physics in 1967. Today, more than three decades later, his paradigm-shifting ideas are becoming popular.

In the 1940s cybernetics began as a formal approach to the description of any physical system in a framework that specified the boundaries, information flows, and goals. Cybernetics has developed the means to model and apply metrics to the otherwise subjective human activities of conversation and collaboration. By enhancing the models of task analysis, workflow, and interaction design, cybernetics provides a toolset for designing successful and human-focused systems.

Edsger Wybe Dijkstra clearly stated that learning physics was real turning point in his comprehension of the discipline of computing. Many theories have been developed independently and what is in practice is a narrow intersection of a few of them. Software Cybernetics being proposed in this paper is a more comprehensive integration of some of these theories.

## II. ARCHITECTING THE AUTOMATON

The word "automaton" is the latinization of the Greek automaton, (neuter) "acting of one's own will". This word was first used by Homer to describe automatic door opening, or automatic movement of wheeled tripods. An automaton (plural: automata or automatons) is a self-operating machine / artifact.

An automaton is a model of computation which

- Executes a step - by - step procedure / method pertaining to a problem specification
- Is operated using a pre-defined set of allowable operations used in computation and their respective costs
- Has a behavior of the whole system resulting from the behavior of each of its components.
- Has some components in the whole system that indicate or specify the "Ground Reality" / Limits.

During the times of Norbert Wiener, servomechanisms in industry, target seeking missiles in the military, and problems of coordinating the war effort, not to forget the emerging mechanization of computation, demanded new vocabularies to understand them. It was no surprise that Wiener defined cybernetics as a science, the science of control and communication.

However, it was engineers who were the first to problematize cybernetic phenomena. They also were the first to use the cybernetic theories and principles as a way to extend their ability to design them. It was Claude Elwood Shannon who postulated that where communication was unambiguous, transmitting a message was an 'engineering problem'.

A well-engineered automaton is expressed using the following mathematical foundations.

1. Complex Numbers and Geometry of Complex Plane
2. Holomorphic Functions and their representation
3. Harmonic Functions
4. Residue Theorem
5. Fourier Transform
6. Anti-Foundation Axiom

This is the mathematics for designing and developing the evolving complex software systems.

“We all realize that we're kind of surrounded with technology: there's little device here recording us, there's tables, chairs, spoons, light bulbs. Each of these things seem pretty mechanical, pretty inert in a certain sense, not very interactive, you know, a hammer, roads. But each one of these technologies actually requires many other technologies to make and produce. So your little thing in your pocket that you use for a phone might require thousands of other technologies to create it and support it and keep it going, and each of those technologies may require hundreds of thousands of subtechnologies below it. And that network of different technologies and the co-dependency that each of those technologies have on each other forms a virtual organism, a super organism.

We can keep stepping back and realize that all these technologies are in some ways co-dependent and related and connected to each other in some way and that largest of all the networks of all these technologies together I call the Technium.”

– Kevin Kelly

The author defines the technium as a Cyber – Physical Systems. An “automaton” models a given thing in the Cyber – Physical System [5]. The most fundamental concept in cybernetics is that of “difference,” either that two things are recognizably different or that one thing has changed over time. The term “differential” is used in calculus to refer to an infinitesimal (infinitely small) change in some varying quantity. If  $x$  is a variable, then a change in the value of  $x$  is often denoted as “delta  $x$  [ $dx$ ]”.

Infinitesimal quantities play a significant role in calculus. Archimedes used them, even though he didn't believe that arguments involving infinitesimals were rigorous. Isaac Newton referred to them as fluxions. Gottfried Leibniz called them differentials and introduced the notation that is used today. In this notation, the differential “ $dx$ ” represents an infinitely small change in the variable  $x$ . Differentials are also used in the notation for integrals. The area under a graph is obtained by subdividing the graph into infinitely thin strips and summing their areas.

Analog computers [13,18] were the first to model the continuously changeable aspects of physical phenomena such as electrical, mechanical, or hydraulic quantities. Setting up an analog computer requires scale factors to be chosen, along with initial condition. Another mandatory effort was to create the required network of interconnections amongst the computing elements. Often times the Cyber – Physical Systems do not lend themselves to the degree of clarity necessary to set up the Analog Computer.

### III. LIMITS IN CYBER – PHYSICAL SYSTEMS

The foundational point of the creation of the Calculus was for an easy mathematical approach for the most reasonable estimation. Trying to use this and its derivative statistical approaches to identify individual entities and their behavior and properties only increases the difficulty exponentially.

Physics has limitations imposed by the very nature of physical reality. Mathematics may provide both real and virtual solutions it is, in and of itself, not limited. Mathematics applied to Physics must be limited, simply by the limits imposed by Physics.

Newton had been approaching calculus primarily in regards to its applications to physics, he purported curves to be the creation of the motion of points while perceiving velocity to be the primary derivative. Conversely, the calculus of Leibniz [12] was applied more to discoveries in geometry made by scholars such as Descartes and Pascal. Since “Leibniz' approach was geometrical,” the notation of the differential calculus and many of the general rules for calculating derivatives are still used today. Leibniz based his calculus on the concept of infinitesimals – positive quantities less than any positive number. Leibniz needed many orders of infinitesimals.

The concept was that of a process (or calculation) whose results were getting ever-smaller so that if inspected, the value of the infinitesimal would be smaller than any named positive real number, and yet the infinitesimal is not zero! There is a curious affinity between infinitesimals and zero numbers, even though they are quite different.

Infinitesimals are not zero, but they adhere to zero. Powers of zero are not zero, but they are not at any distance from zero.

In other words, an infinitesimal is a number whose magnitude exceeds zero but somehow fails to exceed any finite positive number. Although logically problematic, infinitesimals are extremely appealing for modeling continuous phenomena.

Leibniz proposed a "universal character for 'objects of imagination'". He argued that a kind of alphabet of human thoughts can be worked out and that everything can be discovered and judged by a comparison of the letters of this alphabet and an analysis of the words made from them. However, the logicians never constructed Leibniz infinitesimals and proved their being on a line. They only postulated it. The positive infinitesimals are smaller than any positive real number, yet strictly greater than zero. The negative infinitesimals are greater than any negative real number, yet strictly smaller than zero.

The theoretical results of Mathematical Logic proposed by the likes of Church, Turing, Godel et al and practical problems in programming [1] severely limit the mechanization of Intelligence. Often times machinization of solutions for a class of problems is incorrectly deemed similar to automation.

Software Cybernetics is being proposed in this paper to explore:

- the formalization of feedback and self-adaptive control mechanisms in software
- the adaptation of control theory principles to software processes and systems
- the application of software engineering principles and theories to control systems, and
- the integration of the theories of software engineering and control engineering.

This paper positions the notion of computing away from the machine architecture [1] and attempts to capture the forms that the human mind perceives through the calculus of geometry propounded by Leibnitz [2,4]. The mathematics of infinitesimals applied to engineering and technology results in the formalisms such as differential geometry, algebraic geometry, synthetic differential geometry, non-standard analysis and noncommutative geometry. The following aspects of Evolving Complex Software Systems become more amenable to analysis and design.

- Interconnected and interdependent elements and dimensions
- Feedback processes promote and inhibit change within systems
- System characteristics and behaviors emerge from simple rules of interaction
- Nonlinearity
- Sensitivity to initial conditions
- Phase space – the ‘space of the possible’
- Attractors, chaos and the ‘edge of chaos’
- Adaptive agents
- Self-organization
- Co-evolution

Some of the characteristics of such systems are:

- Small inputs can lead to dramatically large consequences
- An infinitesimal change in the initial conditions can produce very different outcomes
- Global properties flow from aggregate behavior of individuals
- Patterns that learn and adapt through interactions with the environment or context of their usage.
- Attractors that result in the states to which the system eventually settles depending on its properties

Limits and infinitesimals enable architecting the “automata” that share the same properties as the original Cyber – Physical System [10,16,17]. The architecture can be visualized through graphics and a grammar of graphics provides insights into the underling deep structure.

#### IV. GRAMMAR OF GRAPHICS

"On a large software project one is lucky if one person in 50 has anything resembling an overall understanding of the conceptual structure of the project, and divinely blessed if that person has the ability to explain it in lay terms"

**- G. Robinson**

Cyber – Physical systems [9] have many autonomous components, i.e., the basic building blocks are the individual agents of the system. There are a large number of useful potential arrangements of their elements. The elements are heterogeneous (differ in important characteristics), i.e. have variety. The system boundary is often hard to pin down. The structure and behavior of a Cyber – Physical System is neither deducible, nor be inferred from the structure and behavior of its component parts. Generally the behavior involves non-deterministic, nonlinear dynamics, sometimes chaos, and rarely any long-run equilibrium. Often times the agents are organized into groups or hierarchies; in which case the structure influences the evolution of the system. Such structures tend to highlight a number of different scales, any of which can affect the behavior of the system. These systems are self-organizing (show a decrease in entropy due to utilizing energy from the environment).

Cyber – Physical Systems adapt to their environment as they evolve. In particular, as they evolve they continually increase their own complexity, given a steady influx of energy (raw resources) and feedback amongst the various components. The components change in response to imposed “pressures” from neighboring components. These systems display emergent macro-level behavior that emerges from the actions and interactions of the individual components. Cyber – Physical Systems can be said to encompass not only one or more system “artifacts” but also the designers and users of these artifacts.

Cyber – Physical Systems are systems that do not have a centralizing authority and are not designed from a known specification, but instead involve disparate stakeholders creating systems that are functional for other purposes and are only brought together because the individual “artifacts” of the system see such cooperation as being beneficial for them.

Understanding how a Cyber – Physical System is difficult primarily because the participants themselves do not understand how it operates. The analysis never starts with the information needed to develop a mathematical representation. Representation of the system as a cognitive model that is verifiable, shared and accurate is not readily amenable to formal mathematical representation. Getting a useful model depends critically upon what is the purpose of the representation and what facets of the system will be analyzed.

Architecting automata that model a Cyber – Physical System [6,7] is becoming increasingly graphical over time. This has resulted in an explosion of visualization techniques and tools that range from basic charting to 3D, big data and Geographical Information Systems visualization. However, all of them deem visualization as a separate entity, and struggle to have provide a common behavior across visualizations and platforms. Adoption of multiple visualization tools across systems further compounds these problems.

The Grammar of Graphics [19, 20] approach allows us to define a common language for all types of visualization. The language approach to specifying visualization provides much more control to adopters while maintaining consistency. The language approach also allows the same specification to be interpreted across platforms to give consistent appearance and interaction across devices.

#### V. INFINITESIMALS, SYMBOLIC GRAPHICS AND SYSTEM DYNAMICS

This paper illustrates the role of infinitesimals in modeling the dynamics of the Cyber – Physical Systems through “Attractors”.

Attractors are patterns in system dynamics. They are usually geometric patterns. It is vital and to represent these geometric patterns symbolically. Symbolic dynamics relates the continuous mathematics of iterations on real and complex spaces to the discrete mathematics of computation and formal linguistics - generically a language of expressing the corresponding automata.

Examples:

- A bracket of Leibnitz structure can describe several dynamical systems with time delay as vector fields associated to smooth functions [11]
- Leibnitz Algebroids [3, 14]
- A Metriplectic (or Leibniz) structure [15]

The evolution of complexity [7] in Nature and Society can be understood as the evolution of the complex software system. Infinitesimals provide a way of modeling the natural automata based on the hierarchy of natural systems from stones and plants up to animals and humans. The information and work flows in such systems that range from stable, oscillating, chaotic and random can be distinguished methodically.

There have been some recent efforts in specifying languages for programming with Infinitesimals. “WHILE Language” for Hybrid System Modeling [8] is one example.

The author is working on Anti-Foundation Axiom, Literate Programming and Software Aesthetics as two best practices that may result in deciding early the beautiful solutions that can be verified and validated.

## VI. CONCLUSIONS

This paper proposes a mathematical approach to architect the automata that model the Cyber – Physical Systems and hence paving way for developing better software systems for the 14 Grand Challenges in Engineering [21] mentioned below.

1. Make solar energy economical
2. Provide energy from fusion
3. Develop carbon sequestration methods
4. Manage the nitrogen cycle
5. Provide access to clean water
6. Restore and improve urban infrastructure
7. Advance health informatics and practice of Medicine
8. Engineer better medicines
9. Reverse-engineer the brain
10. Prevent nuclear terror
11. Secure cyberspace with methods better than the traditional practices
12. Enhance virtual reality
13. Advance personalized learning
14. Engineer the tools of scientific discovery

## ACKNOWLEDGEMENTS

The author places on record his sincere thanks to Anna University, Chennai, the members of the Steering Committee of the Series of Annual Conferences on Theory and Applications of Models of Computation [TAMC] and the Conference Chairs of the IEEE Series of Conferences on Norbert Wiener in the 21<sup>st</sup> Century.

## REFERENCES

- [1] DAVID Evans, Introduction to Computing Explorations in Language, Logic, and Machines, CreateSpace Independent Publishing Platform, 2011.

- [2] DAVID Tall, Infinitesimals constructed algebraically and interpreted geometrically, *Mathematical Education for Teaching*, Vol. 4, No. 1 Pp. 34-53, 1981.
- [3] GHEORGHE Ivan and DUMITRU Opri, Dynamical systems on Leibniz algebroids, *Differential Geometry - Dynamical Systems*, Vol.8, 2006, pp. 127-137.
- [4] JOEL A. Tropp, Infinitesimals: History & Application, Plan II Honors Program, WCH 4.104, The University of Texas at Austin, US, 1997.
- [5] IVAN Ruchkin, BRADLEY Schmerl and DAVID Garlan, Analytic Dependency Loops in Architectural Models of Cyber-Physical Systems, 8th International Workshop on Model-based Architecting of Cyber-Physical and Embedded Systems (ACES-MB), September 28, 2015.
- [6] KATHERINE Brading, On composite systems: Descartes, Newton, and the law-constitutive approach, University of Notre Dame, 2010.
- [7] KLAUS Mainzer, *Thinking in Complexity*, Springer, 2007.
- [8] KOHEI Suenaga and ICHIRO Hasuo, Programming with Infinitesimals: A WHILE-Language for Hybrid System Modeling, Automata, Languages and Programming, Volume 6756 of the series Lecture Notes in Computer Science pp 392-403, 2011.
- [9] LEE E. A and SESHIA S. A, Introduction to Embedded Systems - A Cyber-Physical Systems Approach, LeeSeshia.org, 2011.
- [10] LEONARDO de Moural and GRANT Olney Passmore, Computation in Real Closed Infinitesimal and Transcendental Extensions of the Rationals, Report supported by EPSRC [grant numbers EP/I011005/1 and EP/I010335/1], 2010.
- [11] LORENZO Magnani and RICCARDO Dossena, Perceiving the Infinite and the Infinitesimal World: Unveiling and Optical Diagrams in the Construction of Mathematical Concepts, Proceedings of the 25th Annual Meeting of the Cognitive Science Society, Richard Alterman and David Kirsh (Eds), 2003.
- [12] LYNDON H. LaRouche, Jr, What Leibniz Intended, *EIR Dynamics*, Pp 44 – 49, January 8, 2010
- [13] MAXWELL C Gilliland, *Handbook of Analog Computation*, Systron – Donner Corporation, California, USA, 1967.
- [14] NAMBIAR K K, Intuitive Set Theory, *Computers and Mathematics with Applications*, Vol. 39, No. 1-2, 1999, Pp 183–185.
- [15] PARTHA Guha, Metriplectic structure, Leibniz dynamics and dissipative systems, *Journal of Mathematical Analysis and Applications*, Volume 326, Issue 1, 1 February 2007, Pp 121–136.
- [16] PAUL Glotfelter, TRAVIS Eichelberger, and PATRICK J. Martin, PhysiCloud: A Cloud-Computing Framework for Programming Cyber-Physical Systems, Report supported by US National Science Foundation through grant CNS-1239221, 2014.
- [17] RAGUNATHAN (Raj) Rajkumar et al., Cyber-Physical Systems: The Next Computing Revolution, Design Automation Conference 2010, Anaheim, California, USA, 2010.
- [18] RAJARAMAN V, *Analog Computation and Simulation*, Prentice-Hall of India, 1971.
- [19] WICKHAM Hadley, A Layered Grammar of Graphics, Report Supported by the National Science Foundation under grant 0706949, 2008
- [20] WILKINSON L, *The Grammar of Graphics* (2nd ed.), Statistics and Computing, New York: Springer, 2005.
- [21] WILLIAM Perry et al., Grand Challenges in Engineering, National Academy of Engineering, Report, 2008.